# Powering IR with R

Thomas Jay Benjamin

Lynzee Murray

2023-10-19

## Table of contents

Preface		3
1	Software setup and overview	4
	1.1 Installation	4
	1.2 Using the R Terminal	4
	1.3 Using an IDE	4
	1.4 Installing R packages	4
	1.5 Writing an R script	5
	Exersices	5
	Extras	5
	Keeping R up-to-date	5
	R Profile and Environment	6
2	Importing and cleaning data	8
	2.1 Importing data	8
	2.1.1 From a local file	8
	2.1.2 From an online file	8
	2.1.3 From a database	8
	2.2 Tidy data	8
3	Statistical analysis & data visualizations	9
4	Building parameterized reports	10
5	Collaborating on Code 5.1 Setting up Git	<b>11</b> 11
References		12

### **Preface**

This online reference guide has been developed to accompany the October 19, 2023 workshop at the Ohio Association for Institutional Research and Planning.

The workshop will follow a timeline aligned to the sections of this reference guide:

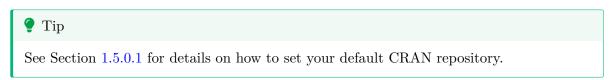
- Chapter 1: Software setup and overview
- Chapter 2: Importing and cleaning data
- Chapter 3: Statistical analyses & data visualizations
- Chapter 4: Building parameterized reports
- Chapter 5: Collaborating on code

### 1 Software setup and overview

- 1.1 Installation
- 1.2 Using the R Terminal
- 1.3 Using an IDE
- 1.4 Installing R packages



The USA (OH) [https] CRAN mirror is hosted by Case Western Reserve University and would be the most suitable for Institutional Researchers working in Ohio (due to its proximity). Its web address is https://cran.case.edu.



### 1.5 Writing an R script

#### **Exersices**

#### **Extras**

### Keeping R up-to-date

#### 1.5.0.1 Updating R packages

In the terminal, run:

```
update.packages()
```

Packages with new updates will be detected and you will be prompted to respond Y or yes to accept.

If you want all packages updated without your confirmation, you can add the ask = FALSE argument:

```
update.packages(ask = FALSE)
```



#### Warning

You will see a warning when packages are built under a different version of R than you are running. If you see this, you likely want to update your version of R.

### 1.5.0.2 Updating R

The installr package can assist with keeping the R installation up-to-date. In the terminal, run the following:

installr::updateR()



You will be prompted as to whether you'd like to copy over all packages from the current version of R. This is a good idea!

#### R Profile and Environment

R will read certain options from special files that you can modify:

#### 1.5.0.1 R profile

Your R profile can contain a range of settings to customize how you interact with R.

The usethis package can be used to edit your R profile:

```
usethis::edit_r_profile()
```

A new window will open with your R profile file that you can edit.

For example, you may want to set your default CRAN mirror to the one hosted at Case, since it is in Ohio and you may be as well. Add the following line to your R profile file:

```
options(repos=c(CRAN="https://cran.case.edu/"))
```

When done, save the file, then close all R terminals and reopen them.

#### 1.5.0.2 R environment

The R environment can store variables that you can call upon using special R functions. It is a great way to keep keys and other secrets out of your code but still accessible.

The usethis package can be used to easily edit your R environment:

```
usethis::edit_r_envrion()
```

To store a variable named test\_var, add the following to the file:

```
test_var:"this is my test value"
```

When done, save the file, then close all R terminals and reopen them.

You'll be able to access such variables like so:

```
Sys.getenv("test_var")
```

[1] "this is my test value"

### ▲ Warning

Some packages request that you store things like API keys as specified environment variables. You can also use this to store common URLs, including FTP sites, but keep in mind that they are accessible directly in the .Renviron file on your computer.

# 2 Importing and cleaning data

- 2.1 Importing data
- 2.1.1 From a local file
- 2.1.2 From an online file
- 2.1.3 From a database
- 2.2 Tidy data

# 3 Statistical analysis & data visualizations

# 4 Building parameterized reports

This will require the rmarkdown and knitr packages.

# 5 Collaborating on Code

### 5.1 Setting up Git

```
git config --global user.email "you@example.com"
git config --global user.name "Your Name"
```

## References